**Sakila Challenge;**

1. List all actors.

SELECT first\_name, last\_name from actor;

1. Find the surname of the actor with the forename 'John'

SELECT last\_name FROM actor where first\_name = “John”;

1. Find all actors with surname 'Neeson'

SELECT first\_name, last\_name FROM actor where last\_name = "Neeson" Order by last\_name ASC;

1. 4. Find all actors with ID numbers divisible by 10.

SELECT \* FROM actor WHERE (actor\_id % 10) = 0;

1. 5. What is the description of the movie with an ID of 100?

SELECT description FROM film WHERE film\_id = 100;

1. Find every R-rated movie.

SELECT title FROM film WHERE rating = "r";

1. Find every non-R-rated movie.

SELECT title FROM film WHERE rating = "r";

1. Find the ten shortest movies.

SELECT title, length from film ORDER BY length ASC LIMIT 10;

OR

SELECT title, MIN(length) FROM film GROUP BY title ORDER BY length ASC LIMIT 10;

1. Find the movies with the longest runtime, without using LIMIT .

SELECT title, length from film ORDER BY length DESC;

OR

SELECT title, MAX(length) FROM film GROUP BY title ORDER BY length DESC;

1. Find all movies that have deleted scenes

SELECT title from film WHERE special\_features = "Deleted Scenes";

1. Using HAVING , reverse-alphabetically list the last names that are not repeated.

SELECT last\_name FROM actor GROUP BY last\_name HAVING COUNT(DISTINCT last\_name) ORDER BY last\_name DESC;

1. Using HAVING , list the last names that appear more than once, from highest to lowest frequency.

SELECT last\_name FROM actor GROUP BY last\_name HAVING COUNT(\*) > 1 ORDER BY COUNT(last\_name) DESC;

1. Which actor has appeared in the most films?

SELECT first\_name, last\_name from actor

Inner JOIN film\_actor ON actor.actor\_id = film\_actor.actor\_id

GROUP BY actor.actor\_id

ORDER BY count(film\_actor.actor\_id) DESC LIMIT 1;

OR

SELECT CONCAT(" ",actor.first\_name, " ", actor.last\_name) AS Full\_name, COUNT(actor.actor\_id) as No\_of\_Movies

FROM actor Inner JOIN film\_actor ON actor.actor\_id = film\_actor.actor\_id

GROUP BY actor.actor\_id

ORDER BY count(film\_actor.actor\_id) DESC LIMIT 1;

1. 'Academy Dinosaur' has been rented out, when is it due to be returned?

SELECT rental\_date, rental\_date + interval (select rental\_duration FROM film where film\_id = 1) day AS due\_date FROM rental WHERE rental\_id = (select rental\_id from rental order by rental\_id desc limit 1);

1. What is the average runtime of all films?

SELECT AVG(length) from film;

1. List the average runtime for every film category.

SELECT category.name, avg(length) FROM film JOIN film\_category using (film\_id) JOIN category using (category\_id) GROUP BY category.name ORDER BY AVG(length) DESC;

1. List all movies featuring a robot.

SELECT title FROM film WHERE description LIKE "%robot%";

1. How many movies were released in 2010?

SELECT COUNT(title) AS No\_of\_film\_released from film WHERE release\_year = 2010;

1. Find the titles of all the horror movies.

SELECT title FROM film\_list WHERE category = "horror";

1. List the full name of the staff member with the ID of 2.

SELECT CONCAT (first\_name, " ", last\_name) AS Full\_name, staff\_id from staff WHERE Staff\_id = 2;

1. List all the movies that Fred Costner has appeared in.

SELECT title FROM film\_list WHERE actors LIKE "%Fred Costner%";

1. How many distinct countries are there?

SELECT COUNT(DISTINCT country) AS No\_of\_Countries from country;

1. List the name of every language in reverse-alphabetical order.

SELECT name from language ORDER BY name DESC;

1. List the full names of every actor whose surname ends with '-son' in alphabetical order by their forename.

SELECT CONCAT (first\_name, " ",last\_name) AS FULL\_NAME FROM actor WHERE last\_name LIKE "%son" ORDER BY FULL\_NAME ASC;

1. Which category contains the most films?
2. SELECT category.name

FROM category

JOIN film\_category ON film\_category.category\_id=category.category\_id

GROUP BY category.category\_id

ORDER BY COUNT(film\_category.category\_id) DESC

LIMIT 1;